OTHER PUBLICATIONS

Kenworthy, "Modems. (Optimizing modem data flow-)(World–Class Tune–Up Tips: Hardware Heaven)", CMP Publications Inc., Windows Magazine, v6,n7, pp. 150–152, Jun. 20, 1995.

[57] **ABSTRACT**

An apparatus for and method of implementing a novel buffer ba full duplex communication system is disclosed. The disclosed invention is particularly useful in native sign processing systems wherein heavy contention of processor resources typically exist, such as in systems running multi-tasking operating systems. The communication system of the present invention includes a receiver, transmitter, echo canceler, CODEC and telephone hybrid. The major components of the system operate on a buffer of input samples consisting of a set of input bits. The communication system operates to generate a buffer of output samples consisting of a set of output bits. The invention utilizes a novel buffer switching mechanism to optimize the tradeoff between processing response time, on one hand, and robustness to interrupt latency and processor implementation on the other hand. The internal processing of the modem works on a buffer full of samples once every time slice thus reducing the probability of a buffer underrun/overrun error occurring. The reduction in probability of data underrun/overrun is achieved by increasing the buffer size, thus giving the operating system greater leeway in choosing the exact time the signal processing functions are run. Small buffers, however, provide the communication system with short and accurate response time. These contradicting motives lead to the novel switchable size buffer scheme of the present invention. This is achieved without a loss of signal coherency.

**12 Claims, 5 Drawing Sheets**

# United States Patent [19]

## Tal et al.

## [56] References Cited

### U.S. PATENT DOCUMENTS

| | | | | |
|---|---|---|---|---|
| 4,841,526 | 6/1989 | Wilson et al. | ............................... | 371/32 |
| 5,077,550 | 12/1991 | Cormier | ................................... | 340/578 |
| 5,327,428 | 7/1994 | Van As et al. | ......................... | 370/94.2 |
| 5,440,740 | 8/1995 | Chen et al. | ............................... | 395/674 |
| 5,585,803 | 12/1996 | Miura et al. | ............................. | 342/372 |

# COMMUNICATION SYSTEM WHICH DYNAMICALLY SWITCHES SIZES OF SAMPLE BUFFER BETWEEN FIRST SIZE FOR QUICK RESPONSE TIME AND SECOND SIZE FOR ROBUSTNESS TO INTERRUPT LATENCY

## FIELD OF THE INVENTION

The present invention relates to communication systems and in particular to modems utilizing native signal processing.

## BACKGROUND OF THE INVENTION

Traditional full duplex communication systems art typically implemented using dedicated hardware. such the prior at modem 12 illustrated in FIG. 1. Modem 12 includes a digital signal processor (DSP) integrated circuit (IC) 18 at its core. a random access memory (RAMS) 20, digital to analog converter (D/A) 24, analog to digital converter (A/D) 26 and data access arrangement (DAA) 28. Modem 12 is shown coupled to a host computer bus through bus interface circuitry 16. Host computer central processing unit (CPU) 14 generates the data to be transmitted through modem 12 and receives the data received by modem 12.

DSP based moderns such as the type illustrated in FIG. 1 generally operate by processing a relatively small number of samples from the input A/D converter 26 and generating a similarly small number of samples to be output through D/A converter 24. These systems are said to work on a 'sample by sample' basis or a 'symbol by symbol' basis. depending on the particular implementation. Such schemes have the benefit of minimal latency time. since the input is processed almost immediately. These schemes. however require very high computing power which is typically provided by a dedicated DSP.

Although modems such as the one illustrated in FIG. 1 can be veer powerful. they are also relatively expensive due to the dedicated DSP hardware involved. Therefore. the recent trend in the personal computer industry is to implement modems by exploiting the built in general CPU without the need for additional processing hardware. This requires the modem to share CPU time with other tasks executed by the system on the same processing unit. In i typical computer CPU. sharing is managed by an operating system on a 'time slice' basis. During every slice a different task executes. Task execution cm be preempted through interrupts caused by a periodic timer indicating the end of we current time slice. or by some other hardware device (e.g.. an AID convener) indicating the occurrence of some event (e.g.. the availability of a sample of data).

Although the interrupt mechanism is designed to switch the currently running task to a task that handles the interrupt in as little time as possible. in practice it may take a substantial amount of time before an interrupt is actually handled. The time delay may be caused by hardware delays. multiple sources of interrupts in the computer system or priority given to some interrupts over others. The time between the actual occurrence of the interrupt trigger and the begs of execution of the interrupt hailer routine is defined as 'interrupt latency.' A typical interrupt latency in the PC environment is in the range of 0 to 5 milliseconds. However. In the PC environment. there is no guarantee that a running program will give up control within a predefined amount of time. Well behaved programs can be expected not to dominate the CPU for an unreasonable time period. however some tasks may hold the CPU resource for a relatively long period of time.

Therefore a straight forward translation of a traditional DSP based modem to a native processing environment (NSP) is very problematic, since it requires the execution of the modem task each sample (or symbol) and the completion of its execution before the next sample (symbol) arrives in order to meet the 'real time' operation requirement. In order to mninimize the time between a sample (or symbol) arrival and the modem activation, a straight forward implementation would be to generate an interrupt upon the arrival of each sample (i.e., symbol). In such an implementation, the real time constraint may be too difficult to overcome and the modem routine may not be executed on time due to long interrupt latencies, which may result in data loss. This problem can be overcome by designing the modem routine to operate on a buffer of samples rather than on one sample only. A buffer of samples means a longer time period between consecutive calls to the modem routine. The real tine requirement in this case is that the time to process m input buffer of samples and to generate an output buffer for transmission is smaller than the time it takes to receive/ transmit a buffer.

The buffer operation scheme, however, poses a new problem. It suffers from an inherent delay disadvantage, since a sample received at the beginning of a buffer is processed only after a whole buffer is received. This disadvantage conflicts with some high rate data pumping modem standards, such as the ITU V.32bis 14.400 bps modem standard, which impose strict time constraints for processing the samples and responding to certain signals from the modem located on the other end of the connection. For example, the V.32bis stank contains a ranging stage at the beginning of the modem connection. During this stage, the specification requires a maximum response time of 26.6 ms to respond to the other modem's ranging signal. The minimum turn around time from the signal detection to response transmission must take into account sample acquisltion, processing, interrupt latency and buffer transmission.

## SUMMARY OF THE INVENTION

The present invention has been developed to alleviate the problems discussed above in the implementation of an NSP based full duplex communication system. These problems include the necessity to be resistant to interrupt operating system and other task related latencies, in addition to opting and being able to operate with to implementation overhead, etc. As discussed above, severe time constraints exist in implementing modems for the public switched telephone network (PSTN) as the transmission bit rate is pushed higher and higher. In most communication systems, the time constraints are typically present during the initial startup of the communication link. In particular, during the ranging phase of the modem connection.

One solution to designing NSP modems capable of implementing higher bit rate standards, e.g., V.32 and V.34, taught by the present invention, is to utilize within the NSP modem buffers of non-fixed size. Allowing the buffer size to vary, allows the NSP software to adapt to the constraints of the standard. Small buffers provide the communication system with short and accurate response times. On the other hand, increasing the buffer size would make the modem processing more robust, reducing the implementation overhead making it more resistant to operating system latencies and giving it greater flexibility in the exact time within the time slice, i.e. between interrupts, that processing can occur without disturbing the data flow. A more robust task is less susceptible to the tasks that are not well behaved, i.e., they hold the CPU for a relatively long period of time.

When the system is in a steady state and can got by with longer response times. it should be able to operate with larger buffers. In such a system. a point in time is reached where the buffer size can be increased without data overruns/underruns or other errors occurring.

In addition. the buffer switching taught by the present invention docs not incur any data loss. Switching occurs smoothly and coherently without the loss of any data. Coherency is hereby defied as not losing or towing away any input or output samples.

Accordingly. it is an object of the present invention to provide a system for enabling high bite rate NSP based communications thereby overcoming the problems associated with the prior art.

It is another object of the present invention to provide a system that can vary the size of its data buffers in accordance with the desired latency time period.

Yet another object of the present invention to provide a system that can vary the size of its data buffers without incurring any data overruns or other data errors.

Another object of the present invention is to provide a system which optimizes system resources such as CPU usage. by reducing the relative implementation overhead of a call to the communication signal processing process.

It is yet another object of the present invention to provide a system which optimizes processor usage by using block optimized signal processing techniques which are more efficient for long data blocks.

The present invention discloses a full duplex communication system that utilizes a buffer interface for both signal reception and transmittal. The buffer interface includes a mechanism of switching between buffers of different sizes without losing any input or output samples. The buffer switching based communication system of the present invention collects a group of input samples and places them in a buffer. The system then processes the samples contained in the buffer and generates a buffer of samples to be transmitted. The reception and transmission buffers are typically of the same length. The length of the sample buffer determines the memory that must be allocated to store the buffer and it also determines the latency time. The latency is defined as the minimum communication system response time between the occurrence of an event at an input port to the generation of another event on an output port in response to the event at the input port. The worst case latency is shown to be exactly twice the time it takes to fill a buffer.

During the initial startup sequence a relatively short and accurate response time is required. Thus. during this phase. short buffers are used. However. later on in the long run (i.e.. in the steady state). long buffers become more efficient. Therefore. there exists some point in time when the modem switches to a buffer with a different size. The buffer switching occurs without losing any coherency (i.e. without losing any data at the input or output ports).

There is thus provided. in accordance with a preferred embodiment. a method. in a communications system of achieving a balance between processing response time. on the one hand. and robustness to interrupt latency and processor implementation overhead. on the other hand. the method including of the steps of utilizing data buffers having a first buffer size when it is desired to optimize the communication system so as to have quick processing response times. and utilizing data buffers having a second buffer size when it is desired to optimize the communication system so as to be robust to interrupt latency and to have low processor implementation overhead.

In addition, the method other includes the step of providing switching means enabling the communication system to switch between using the buffers having a first buffer size and the buffers having a second buffer size. Also, the size of the buffers is coherently switched without any loss of data and the second buffer size is greater than the first buffer size. In addition, the second buffer size is switched back to a smaller size when modem connection is reinitialized or restarted. The second buffer size can be switched back to a smaller size when a retrain sequence has been initialized, wherein the communication system implements an International Telecommunication Union standard chosen from the group of V.32, V.32bis an V.34.

There is also provided, in accordance with a preferred embodiment of the present invention, a method of implementing a communications system the communications system comprising a transmitter, echo canceler and a receiver, the method including the steps of perfoming echo cancellation, utilizing the echo canceler, on delayed output samples transmitted during time slice K–1, performing receive processing, utilizing the receiver, on the difference between input samples received from time slice K–1 and samples generated by the echo canceler during time slice K, an performing transmit processing, utilizing the transmitter, to generate the output samples to be transmitted during time slice K+1.

In addition, the transmitter, the echo canceler and the receiver are implemented using a central processing unit (CPU) of a computer and the size of the buffers is coherently switches without any loss of data.

## BRIEF DESCRIPTION OF THE DRAWINGS

The invention is herein described, by way of example only, with reference to the accompanying drawing;, wherein:

FIG. 1 is a high level block diagram illustrating the a prior art conventional digital signal processor (DSP) modem coupled to a host computer;

FIG. 2 is a high level functional block diagram illustrating a functional view of a modem utilized in the communication system of the present invention;

FIG. 3 is a high level block diagram illustrating the native signal processing NSP) modem of the present invention coupled to a host computer;

FIG. 4 illustrates the sequence of signals exchanged by the calling and answering modems during the start-up procedure of the V.32 bis modem standard;

FIG. 5 illustrates the steps of receiving, processing and transmitting in relation to the sequencing of time slices; and

FIG. 6 is a high level flow diagram illustrating the processing sequence of the present invention.

## DETAILED DESCRIPTION OF THE INVENTION

To better illustrate the operation and utility of the buffer switching system of the present invention, the system Is described in the framework of a full duplex voiceband modem. However, it is understood that the example presented throughout this disclosure in no way limits the scope of the present invention. One skilled in the art may take the principles of the system and methods of the present invention disclosed herein and apply them to many other types of full duplex communication systems, those of which that are well known in the art.

A high level functional block diagram illustrating a general realization of a full duplex voiceband modem 30

utilized in the communication system of the present invention is shown in FIG. 2. Modem 30 contains a transmitter 32, a receiver 34, an echo canceler unit 36, a summer 37, a digital to analog (D/A) convener 38, an analog to digital (A/D) convener 40 and a digital access arrangement (DAA) or hybrid 42. Echo canceler unit 36 comprises delay register stack 35 and echo canceler circuitry 41. Transmitter 32 received data from the transmitter data in port and outputs transmit (Tx) samples to echo canceler 36 and D/A 38. DAA 42 functions to match the impedances between the telephone line and the transmitter and receiver. It transforms balanced analog voltage On the two-wire pair from the central office (CO) to two two-wire unbalanced pairs, one for the transmitter and one for the receiver. Echo canceler 36 functions to remove echoes from the received signal by applying standard echo canceling techniques, which are well known in the art, to the transmitted signal. The output echo canceler 36 is summed with the receive signal using summer 37. Receiver 34 outputs a digital receive (Rx) data out signal.

A high level block diagram illustrating the native signal processing (NSP) modem, generally referenced 10, coupled to a host computer is shown In FIG. 3. NSP modem 10 generally comprises a hardware portion and a software portion. The software portion runs on host computer CPU 54. FIG. 3 illustrated an implementation of a full duplex voiceband modem where the signal processing tasks shown in FIG. 2, i.e., transmitter 32, receiver 34 and echo canceler 36, are performed using a host computer central processing unit (CPU) 54. The host computer also includes random access memory (RAM) 52 for use by the operating system of the host computer and its tasks. During execution of the NSP modem, portions of the software used to run the modem are resident at varying times within RAM 52.

A DAA 64 forms the physical line interface to the 2-wire pair from the CO (e.g. RJ-11, RJ-45 or any other suitable connection method). The host CPU 54 communicates to NSP modem 10 through bus interface circuitry 56. Two first in first out (FIFO) buffers are used to buffer samples to and from the host computer CPU. A transmit FIFO 58 buffers outbound samples and a receive FIFO 60 buffers inbound samples. A coder/decoder (CODEC) 62 couples a transmit FIFO 58 and receive FIFO 60 to DAA 64. CODEC 62 performs the D/A and A/D functions of D/A 24 and A/D 26 (FIG. 2).

Since the majority of personal computers today run some type of muli-tasking operating system. It will be assumed that host computer CPU 54 is executing some form of mult-tasking operating system. In this case, the system of using buffers of varying size taught by the present invention is used by NSP modem 10 to conform to the strict time tolerances imposed by the modem standards, e.g., V.32 and V.34 standards. To aid in understanding the mechanism of using buffers of varying sizes as taught by the present invention, the V.32 bis standard is used as an example. In particular, the V.32 bis start-up procedure which occurs before data can be exchanged, is described. Illustrated in FIG. 4 is the stat-up sequence performed by the calling and answering modems as dined in the V.32 bis standard. The top portion shows the transmitted signal of the calling modem (i.e., the modem that placed the call) as a function of time. The bottom portion shows the tasks performed by the answering modem as a function of time. During the initial sequence, up until the 'S' on the answering modem, very short and accurate response times are required (e.g., 64T+/−2T which translates to 26.6+/−0.83 ms). During this so called ranging period, the round trip time delay of the far and echo is measured. These measurements are then used in the

echo canceler circuitry **41** to eliminate echoes from the received signal.

To respond within the allotted time frame, short response times are needed. Thus, during this initial phase, short buffers are used. After the training stage 'TRN' onward, short and accurate event handling in not an absolute necessity and long buffers may be utilized, thus reducing the CPU context switching task overhead. Experiments undertaken by the inventor, using a PC equipped with a Pentium 100 MHz processor, have shown hat CPU utilization during the ranging period is relatively low, on the order of less than five percent During the training period CPU utilization increases add con exceed thirty percent. During the steady state data portion CPU utilization falls to less than thirty percent.

FIG. 5 illustrates the steps of receiving, processing and transmitting in relation to the sequencing of time slices. Each column, located between two vertical lines represents one time slice. The vertical lines represent hardware interrupts that occur once at the start of each time slice or sample time. For each time slice the communication process running within NSF modem **10** is executed. During this time, the samples that were collected during the previous time slice are processed and samples are generated that are to be transmitted during the next time slice. Thus, samples that are collected during time slice N-4 are processed during time slice N-3. Also, during time slice N-3, samples are generated which will be transmitted during time slice N-2.

As previously discussed, during the ranging period of the start-up sequence (FIG. 4) both modems have a time constraint of 26.6+/−0.83 ms (i.c., 64T+/−2T) in which to respond to the other's signal. Assuming an average processing and associated hardware delay of 6.6 ms, there remains only 20 ms for modem processing, i.e., 20 ms for the inherent delay in the buffer mechanism. The maximum or worst cue delay in the buffer mechanism is two buffer let, given the buffer processing scheme described above. Hence, the maximum buffer length in time, can not exceed 10 ms. Thus, for CODECs that have a sampling frequency of 8 KHz, buffers having a length L1 equal to 64 samples are used during ranging. This translates to a buffer time of 8 ms. Assuming a processing overhead of 3 ms leaves a margin of 5 ms. Thus, if the interrupt latency or any other operating system imposed delays is greater than 5 ms, a data overrun will occur or the ranging between the modems will not be accurate enough for communications to take place.

At some point in time, the decision to switch to large buffers is made. In the case of V.32 bis, the switch to large buffers con occur after the ranging period has concluded, preferably before the training period has begun. The system of the present invention is oblivious to the samples contained in the sample buffers. It makes no difference whatsoever what the samples within the buffer are. The coherency characteristic of the present invention is described in more detail below.

The operation of the buffer switching mechanism around the point of switching will now be described in more detail. Assume time slice N-1 is current and a buffer full of samples is received during this slice. Assume also that small buffers are no longer needed and the decision to switch to larger buffers has been made. What entity makes the decision to switch is not relevant to the present invention. During time slice N, the sample received during slice N-1 are processed. However, the processor knows that from the next time slice forward, large buffers are to be used. Thus, the processor generates a buffer of samples to be transmitted that has a length L2 greater than L1. For example purposes L2 is equal

to 256. The transmit process is independent of the other components of the system and therefore can produce buffers having any arbitrary number of samples. Thus, a buffer of size 256 samples is generated by the processor during slice N. These samples will be transmitted during the following time slice N+1. Also during slice N, a small buffer of samples is received that are to be processed during slice N+1.

During the next slice N+1, a large buffer of samples is acquired and the small buffer of samples acquired during slice N is now processed. A large buffer of samples is generated for trasnmission during slice N+2. In addition, the large buffer of samples generated during slice N are transmitted during this slice N+1. Similarly, during slice N+2, the large buffer of samples received during slice N+1 are processed and a large buffer of samples is generated for transmission during slice N+3. The following table shows the buffer sizes used before and after the switching transition.

| Time Slice | Buffer Size Processed by Echo Canceler | Buffer Size Processed by the Receiver | Buffer Size Processed by the Transmitter |
|---|---|---|---|
| <N | L1 | L1 | L1 |
| N | L1 | L1 | L2 |
| N + 1 | L1 | L1 | L2 |
| >N + 1 | L2 | L2 | L2 |

With reference to the table above, the echo canceler processes the received buffer of slice K−1 during slice K, K being any arbitrary time slice. Thus, processing L1 samples in buffers less than or equal to N+1 and outputting buffers of exactly L1 samples. During time slice K, the receive process processes the difference between the buffer received in slice K−1 and the output of the echo canceler in slice K. Thus, in time slice N, the receiver will process the difference between the samples received during slice N−1, having a length L1, and the output of the echo canceler during slice N, also having a length L1. During time slice N+1, the receive process will process the difference between the received samples in time slice N, having a length L1, and the output of the echo canceler during slice N+1, also having a length L1.

To ensure that the receiver gets samples that have had echoes properly removed from them, the transmitted and received samples must be suitably aligned. In other words, the samples used by the echo canceler must be synchronized in time with the samples transmitted and received. This is achieved by placing one buffer delay register 35 (FIG. 2) in the path of the transmitted data before it is used by the echo canceler (assuming echo canceling processing occurs before transmitter processing). Thus, the transmitted data used by the echo canceler is properly delayed so as to be synchronized to the received data. Delay registers 35 ensure, that a small buffer is always subtracted from a small buffer and that a large buffer is always subtracted from a large buffer, thus maintaining data coherency.

In order to meet the time constraints of the modem standards, the processing of the data must occur in a pertain order. Illustrated in FIG. 6, is a high level flow diagram showing the sequence of processing. Echo canceler unit 36 (FIG. 2) must run before the receiver sinus samples which are processed by the receiver comprise the difference between the received sample and the output of the echo canceler. As described previously, a summer 37 performs the